I hereby certify that this paper (along with any paper referred to as being attached or enclosed) is being transmitted via the Office electronic filing system in accordance with § 1.6(a)(4).

Dated: September 8, 2009 Signature: /Jeffrey H. Canfield #38,404/
(Jeffrey H. Canfield)

Docket No.: 06005/39537
(PATENT)

## IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of:
Gary K. Law et al.

Application No.: 10/656,005

Confirmation No.: 8163

Filed: September 5, 2003

Art Unit: 2179

For: STATE MACHINE FUNCTION BLOCK WITH A USER MODIFIABLE STATE TRANSITION CONFIGURATION DATABASE

Examiner: S. B. Theriault

## APPEAL BRIEF

MS Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

As required under § 41.37(a), this brief is filed not more than two months after the Notice of Appeal filed in this case on July 8, 2009, and is in furtherance of said Notice of Appeal.

This brief contains items under the following headings as required by 37 C.F.R. § 41.37 and M.P.E.P. § 1205.2:

## TABLE OF CONTENTS

**Appendices**

## I. REAL PARTY IN INTEREST

FISHER-ROSEMOUNT SYSTEMS, INC. is the real party in interest for this appeal as evidenced by the Assignment recorded in the assignment records of the Office at Reel/Frame number 014721/0881.

## II. RELATED APPEALS AND INTERFERENCES

There are no other appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in this appeal.

## III. STATUS OF CLAIMS

Claims 1-79 are pending in the application. All stand rejected and are presently appealed.

## IV. STATUS OF AMENDMENTS

Applicant did not file an Amendment After Final Rejection.

## V. SUMMARY OF CLAIMED SUBJECT MATTER

**Independent claim 1** calls for a method of configuring a state machine implemented in a function block associated with a process plant. The state machine is configured via a computing device 18a that includes a display device 120 and an input device 124. (See Fig. 2, paragraph [0041]). The state machine defines a plurality of states 904, 908, 912, and the state machine transitions between states based on state machine configuration data and one or more state machine inputs. (Fig. 17, paragraph [0100]). The state machine inputs are associated with operation of the process plant. (Paragraph [0051]). The method comprises providing a graphical user interface 300 to be displayed by a display device associated with the computing device. (Fig. 5, paragraph [0053]). The graphical user interface includes a plurality of graphical elements 304 defining input/state pairs. (Figs. 5 and 6, paragraph [0053]). The method further includes receiving state transition data 304A, 304B, 304C, 304D associated with one or more of the plurality of graphical elements via the input device. (Fig. 6, paragraph [0055]). For each of the one or more of the plurality of graphical elements for which state transition data is received 304A, 304B, 304C, 304D , the state transition data identifies a next state to which the state machine transitions following conditions in the process plant corresponding to the input/state pairs defined by the particular graphical elements. (Fig. 6, paragraph [0055]). Finally the method calls for storing the state transition data on the first computer readable medium 406 associated with the function block (Fig. 8, paragraph [0062]).

**Independent claim 18** calls for a tangible medium storing machine readable instructions. The machine readable instructions include first code for providing a graphical user interface 300 via a display device. (Figs. 5 and 6, paragraph [0053]). The graphical user interface is provided for configuring state machine transitions among a plurality of states 904, 908, 912. (Fig. 17, paragraph [0100]). The graphical user interface includes a plurality of graphical elements representing state machine input/state pairs 304A, 304B, 304C, 304D. (Fig. 6, paragraph [0055]). The plurality of graphical elements 304A, 304B, 304C, 304D can be used to indicate desired transitions between states. Id. The machine instructions further include second code for receiving state transition data identifying state machine next states associated with one or more of the graphical elements. Id. The state machine state transition data may be received via the graphical user interface. Finally, the machine instructions include third code for storing the state transition data on a computer readable medium 406 associated with a function block 404 implementing a state machine in a process plant. (Fig. 8, paragraph [0069]). When the state transition data are stored on the computer readable medium associated with the function block implementing the state machine in the process plant, the state machine transitions from a current state to a next state when the current state and conditions in the process plant correspond to an input/state pair associated with a graphical element. (Paragraph [0069]).

**Independent claim 34** calls for a method of implementing a state machine in a function block for use in controlling or simulating control of one or more field devices in a process plant. (Fig. 1, paragraph [0046]). The method of claim 34 calls for providing a graphical user interface 300 displayed by a display device. (Fig. 5 and 6, paragraph [0053]). The graphical user interface 300 includes a plurality of graphical elements 304 for configuring state machine transitions between a plurality of state machine states. The graphical elements 304 define one or more state machine input/state pairs, wherein one or more state machine inputs represent various conditions within the process plant. (Figs. 5 and 6, paragraph [0053]). The method further calls for receiving state transition data 304A, 304B, 304C identifying a state machine next state associated with at least one of the graphical elements via an interface input 124 associated with the graphical user interface. The method further calls for storing the state transition data on the first computer readable medium associated with the function block (Paragraph [0062]). In operation, the method calls for receiving at least one state machine input and determining a state machine next state based on the at least one input, a current state, and the state transition data stored on the first computer readable medium. When the appropriate next state has been determined, the state machine transitions to the next state by setting the current state of the state machine to the determined next state. (Fig. 7, paragraph [0063]). Finally, the method of claim 34 calls for providing at least one function block output for use in controlling the one or more field devices to at least a second other function block, wherein the at least one function block output is based on the current state of the state machine. (Fig. 7, Paragraph [0064]).

**Independent claim 58** calls for a function block entity for use in a process plant. The process plant includes a processor adapted to control or simulate control of one or more field devices. (Fig. 1, paragraphs [0030-0040]). The function block entity comprises a user modifiable state machine configuration 406 database. The state machine configuration database 406 includes state transition data 304A, 304B, 304C, 304D indicative of how a state machine implemented by a function block is to transition among a plurality of states. (Fig. 6, paragraph [0055]). The state transition data comprises potential pairings of state machine states and one or more corresponding function block inputs (Figs. 5 and 6, paragraph [0053]). The state transition data indicates the next state to which the state machine transitions when the state machine is in a particular state corresponding to a one of the input/state pairings and when an input corresponding to the particular input/state pair is a particular value (Fig. 6, paragraph [0055]). The function block entity further comprises a first computer readable medium 406. First code is stored on the first computer readable medium and is adapted to receive inputs to the function block 404. The inputs comprise data associated with the operation of the process plant. (Paragraph 51). Second code data 404 are stored on the first computer readable medium to determine a next state of the state machine. The next state is determined based on it the at least one input, a current state of the state machine, and the state transition data. (Paragraph [0069]). The second code is fixed. The function block entity further includes third 412 code stored on the first computer readable medium 406 and is adapted to set the current state of the state machine to the next state. Again the third code is fixed. Finally the function block entity includes fourth code 420 stored on the first computer readable medium the fourth code is adapted to provide at least one function block output for use in controlling the one or more field devices. (Fig. 7, paragraph [0064]).

## VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Grounds of Rejection to Be Reviewed on Appeal

Are claims 1, 34 and 58 patentability distinct from claims 1, 35 and 49 of US Patent No. 7,269,468?

Are claims 1-79 anticipated by U.S. Patent No. 6,834,370 to Brandl et al.?

Are claims 1-79 obvious over U.S. Patent No. 6,834,370 to Brandl et al. in view of U.S. Patent No. 5,903,886 to Heimlich et al.?

## VII. ARGUMENT

### A. Claims 1, 34, and 58 are not unpatentable under the judicially created doctrine of obviousness type double patenting over U.S. Patent No. 7,269,468.

A nonstatutory obviousness-type double patenting rejection is appropriate where the conflicting claims are not identical, but at least one examined application claim is not patentably distinct from the reference claims because the examined application claim is either anticipated by, or would have been obvious over, the reference claims. See, e.g., In re Byrd, 140 F.3d 1428, 46 USPQ2d 1226 (Fed. Cir. 1998); In re Goodman, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); In re Longi, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); In re Van Omum, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); In re Vogel, 422 F.2d 438, 164 USPQ 619 (CCPA 1970); and In re Thorington, 418 F.2d 528, 163 USPQ 644 (CCPA 1969). Claims 1, 34, and 58 are patentability distinct from the claims of the '468 patent because none of the claims are obvious in light of any of the claims in U.S. Patent No. 7,269,468.

Claims 1 and 34 of the present application relate to methods of either configuring or implementing a state machine in a function block associated with a process control system in a process plant. Claim 58 of the present application relates to a function block entity for use in a process plant implementing a state machine. Each of the methods called for in claims 1 and 34 include receiving state transition data that identify a "next state" to which the state machine transitions when the state machine is in a particular state and a particular set of input signals are input to the state machine. Similarly, claim 58 calls for a user modifiable state machine configuration database that includes state transition data indicative of how a state machine transitions among a plurality of states. In each case, the state machine transition data comprises input/state pairs, and corresponding state machine "next state" information. The state machine transitions from a current state to an identified next state when the current state of the state machine and the state machine inputs correspond to a particular input/state pair.

The claims of U.S. Patent No. 7,269,468 (hereafter the '468 patent), on the other hand, relate at least in part to the configuration of outputs associated with a state machine implemented within a process control function block. Claim 1 of the '468 patent, for example, calls for providing a first graphical user interface via a display device to configure values of at least some outputs of a plurality of outputs of a function block for some states of a plurality of states of a state machine. Claim 1 further calls for receiving output configuration data via the graphical user interface and storing the output configuration data on a computer readable medium. Independent claim 35 of the '468 patent includes language nearly identical to that of claim 1. Independent claim 45 of the '468 patent calls for a function block implementing a state machine rather than a method of configuring a function block implementing a state machine. Nonetheless, claim 45 calls for a user modifiable state machine configuration database including output configuration data indicative of values of the least some outputs of the plurality of outputs of the function block implementing the state machine.

None of the claims of the '468 patent call for any steps related to, or structure for performing the steps of, configuring the state transitions of a state machine implemented in a function block executed within a process control system of a process plant. In fact, in the Final Office Action the Examiner agrees with this assessment of the differences between the claims of the present application and the claims of the '468 patent. (See Final Office Action, page 2, last 3 lines). The Examiner argues, however, that these differences between the present invention and the claims of the ’468 patent are obvious variations in dealing with and controlling/storing data 10. The Examiner's position, however, simply cannot be supported when one considers the subject matter claimed in the present application versus the subject matter claimed in the ’468 patent.

Configuring the outputs of a state machine involves identifying which outputs will be set or asserted when the state machine is in a particular state. Fig. 6 of the ’468 patent shows an output configuration matrix having various state machine states listed along the vertical axis and a number of discrete outputs listed along the horizontal axis. The matrix defines a plurality of cells, each cell corresponding to a particular state/output pair. The state machine outputs are configured by placing an X in the cells corresponding to outputs that are to be set when the state machine is in the state corresponding to a selected cell. As shown in Fig. 6, an X is placed in the cell 304B. Cell 304B corresponds to State 2-Vent, and Output 3 –OPEN VENT 120. Thus, according to the output configuration data shown in Fig. 6 of the ’468 patent, Output 3 – OPEN VENT 120 – will be asserted when the state machine is in State 2 – VENT.

As the above example shows. All that is required to configure the outputs of a state machine implemented according to the ’468 patent is to identify which outputs are to be asserted during which states. The ’468 does not teach or suggest a mechanism for configuring the state transitions between the various states of the state machine. The ’468 patent does not teach a mechanism for defining the state machine states, nor does it teach a mechanism for defining when or the state machine transitions between the states. Furthermore, the claims of the present application are not mere variants of the invention claimed in the ’468 patent. The claims of the present application define mechanisms and methods for configuring completely different aspects of the operation of a state machine from those taught by the claims of the '468 patent.

Because the claims of the '468 patent do not teach or suggest the invention claimed in the present application and because the claims of the present application are more than mere variants of what is claimed in the '468 patent, the claims of the present application are patentably distinct from the claims of the '468 patent. Since an obviousness type double panting rejection is appropriate only when the pending claims are not patentably distinct from the reference claims, the obviousness type double patenting rejection in the present case is improper and should be overturned.

## B. **Claims 1-79 are not anticipated under 35 U.S.C. §102(e) by U.S. Patent No. 6,834,370 to Brandl *et al.***

A claim is anticipated only if each and every element as set forth in the claim is found either expressly or inherently described in a single prior art reference. Verdegaal Bros. v. Union Oil Co. of California, 814 F.2d 628, 631, 2 USPQ 2d 1051, 1053 (Fed. Cir. 1987). Claims 1-79 are not anticipated by U.S. Patent No. 6,834,370 to Brandl et al. (hereafter "Brandl") because Brandl does not teach each and every element of any of claims 1-79 pending in the present application.

Claim 1 is exemplary. Claim 1 calls for, among other things, providing a graphical user interface displayed by a display device. The graphical user interface includes a plurality of graphical elements defining input/state pairs associated with a state machine. Claim 1 further calls for receiving state transition data associated with one or more of the plurality of graphical elements via the input device, wherein for each of the one or more of the plurality of graphical elements for which state transition data is received the state transition data identifies a next state to which the state machine transitions following conditions in the process plant corresponding to the input/state pair defined by the graphical elements.

Brandl does not teach a user interface that includes a plurality of graphical elements defining input/state pairs. Brandl does not teach receiving state transition data associated with one or more of a plurality of graphical elements via an input device. And Brandl does not teach receiving state transition data that identify a next state to which a state machine transitions following conditions in a process plant corresponding to an input/state pair defined by a particular graphical element.

In the final office action, Examiner cites Brandl column 52, lines 23-35 and column 49, lines 19-25 as well as Figs. 35, 72, 74, 84-86 as teaching a graphical user interface displayed by a display device, the graphical user interface including a plurality of graphical elements defining input/state pairs. A review of the cited passages and corresponding figures, however, reveals that Brandl teaches no such things. At col. 52, lines 23-35, Brandl states:

> Fig. 84 depicts the Sulferize_UOP:1 unit operation which is the only unit operation 52 underlying the Sulferize_UPC:1 unit procedure 50 and FIG. 85 depicts the detailed procedure for the sulferize unit procedure in PFC format. FIG. 86 depicts the detailed procedure for the esterify unit procedure in PFC format and FIG. 87 depicts the detailed procedure for the separate unit procedure in PFC format. FIG. 88 provides a list of the process actions 62 from the general recipe 44 defined by FIGS. 72, 74, 80, and 81 and the recipe segment 64 for the master recipe 46 that correspond to the process actions of the general recipe.

At col. 49, lines 19-25, Brandl states:

> FIG. 72 shows the recipe editor with the Procedure file folder of the general recipe selected in the file folders pane 142 of the editor. The view pane 144 depicts the general recipe 44 depicted as a process dependency chart at the

> highest level, the process stage level. Again, refer to Appendix 2 for a detailed description of how to read a process dependency chart.

Neither of these passages teach anything regarding graphical elements that are displayed on a display device and which define input/state pairs associated with the configuration of a state machine.

The figures cited by the Examiner likewise fail to teach anything regarding graphical elements displayed on a display device defining input/state pairs. Fig. 35 shows "a general recipe of a first example including underlying process actions and process operations." (Brandl, col. 13, lines 57-60.) Fig. 72 shows "A view from the general recipe editor in accordance with preferred embodiments of the invention for the process dependency chart for the general recipe of the second example." (Brandl, col. 15, lines 26-29.). Fig. 74 shows "A view from the general recipe editor in accordance with the preferred embodiments of the invention for process details of the sulfurize process stage for the general recipe of the second example." (Brandl, col. 15, lines 33-36.) Figs. 84-86 show various flowcharts for a unit operation Sulferize_OP: 1. (See Brandl, col. 16, lines 7-16.) None of these drawings has the slightest relevance to providing graphical elements that define input/state pairs associated with the configuration of a state machine.

Next, the Examiner pointed to Figs. 84-86 and 35 as teaching receiving state transition data associated with one or more of a plurality of graphical elements via an input device, where the state transition data identifies a next state to which the state machine transitions following conditions in the process plant corresponding to the input/state pairs defined by the graphical elements. Again, the cited figures teach no such thing!

As mentioned above, Figs. 84-86 are flowcharts depicting a "sulferize" unit operation. (Brandl, col. 16, lines 7-16.) Fig. 35 shows "the general recipe of a first example including the underlying process actions and process operations." (Brandl, col. 13, lines 58-60). There is absolutely no discussion, disclosure, or representation of the step of receiving state transition data associated with one or more of a plurality of graphical elements via an input device in these figures. What is more, there is no teaching of state transition data that identify the next states to which a state machine transitions following conditions in the process plant corresponding to the input/state pairs defined by the graphical elements, as called for in independent claim 1. The examiner has not identified a single state transition disclosed by Brandl that occurs based on operating conditions in a process plant and which is identified in state transition data corresponding to an input/state pair defined by a graphical element displayed in a graphical user interface.

According to the examiner "a state machine models behavior or a process or procedure and the machine transitions between those states, and provides an output. In the common art, the state an refer to logical state, such as 1 or 0. Or, it can refer to "on" and "off" or state can reflect context; meaning operation or status. A button on an interface can be shown as pressed, which reflects state. A graphical element that displays the input of a temperature value along with a motor control value to operate a machine with direct output of the result to the graphical interface to show the temperature increasing and the valve opening is a state changing machine." (Final Office Action, page 13, lines 12-18.) Further, the examiner states that Brandl expressly shows inputs directed to a graphical element that will display a specific output with a given state transition applied to them. (Final Office Action, page 13, lines 19-20). Based on the Examiner's remarks in the Final Office Action it appears that the Examiner fundamentally misunderstands the nature of a "state machine" as claimed in the present application and as would be understood by those of ordinary skill in the art. The Examiner apparently further misunderstands the nature of the configuration information necessary configure such a state machine.

A state machine is a logical construct that may be implemented in hardware or software. A state machine may be configured to take on a number of different logical states. Although the state machine may take on a number of different states, it may occupy only one state at any given time. The state machine may transition from one state to another based on the current state of the state machine and the state of various state machine inputs according to the state machine configuration data. The state machine may provide outputs that are based on the current state of the state machine. (Specification, paragraph [0048]). The configuration data may specify the number of states that the state machine may take on, and may specify how and when the various state machine inputs cause the state machine to transition between the various states. (Specification, paragraph [0052]).

The claims of the present application all relate to mechanisms, either methods or various software entities, by which a programmer may enter configuration information for configuring the state transitions associated with a state machine. Each of the independent claims pending in the application in some way calls for the definition of input/state pairs. As can be seen in Figs. 5 and 6 of the present application, the input/state pairs match various state machine inputs with respective state machine states. The claims further call for either a step of, or provisions for, receiving state transition data corresponding to one or more input/state pairs. The state transition data define a "next state" to which the state machine transitions when the state machine is in a particular state defined by a particular input/state pair and when the input corresponding to the particular input/state pair is asserted. Thus, according to the claims, when the conditions in a process plant corresponding to a particular input/state pair are true, the state machine will transition to a "next state" defined by the configuration data as being associated with the particular input/state pair.

In the Final Office Action the Examiner gives a number of examples which supposedly teach a state machine that is configured by user input to function in a certain manner. The Examiner states: "For example, (fig. 85) Brandl expressly shows in step 64 two inputs or pairs where the charge 1 and the charge b_1 are combined to form temp_ctrl 1: state. The status is reflected that the charges are complete the transition the data to complete to move to the next step in the recipe. Following the steps shown in figure 74, the user can see that the inputs to a given operation can have a single path or dual path, as shown in column 154. Dual paths require input pairs and direct transitions to the next state." (Final Office Action page 13, last paragraph). It is difficult if not impossible to follow the Examiner's arguments as to how Brandl teaches a user interface that includes a plurality of input/state pairs for receiving state transition data that identify a next state to which the state machine transitions following conditions in a process plant corresponding to one of the various input/states pairs. Brandl Fig. 85 merely shows a block diagram of a segment of a master recipe. (Brandl, col. 52, lines 35-34). Apparently the Examiner is equating the completion of the steps Charge 1 and Charge b_1 to inputs to a state machine and considers the inputs Charge 1 and Charge b_1 as an input/state pair associated with a state machine. Charge 1 and Charge1_b may comprise an input pair, but they say nothing regarding a state machine state and they are most certainly not an input/state pair. In fact, Brandl Fig. 85 discloses nothing that may remotely be considered a state machine input/state pair.

Furthermore, Brandl Fig 74 merely shows a process sequence table. The process sequence table is broken up into a group of columns. The leftmost column is the index number associated with the steps and elements of the "sulfurize" procedure. The next column indicates what steps may be performed in parallel, where index numbers are to be performed in parallel, when to start parallels, and when to finish parallels. The next column lists a path number that identifies which path of a parallel a particular index number is associated with. The next column provides a name and action associated with a process step. Another column lists a material associated with a process materials associated with the various process steps. Another column identifies a formula identifier associated with the various process steps, and so forth. (See, col. 49, line 55-Col. 50, line 17). Again, nothing in Brandl Fig. 74 nor the accompanying description describes input/state pairs associated with a state machine. Nor does Brandl Fig. 74 show next state information defining a next state to which a state machine transitions following conditions in a process plant corresponding to a particular input/state pair.

Even a casual reading of Brandl makes it abundantly clear that what is taught by Brandl has absolutely nothing to do with configuring the state transitions of a state machine. Brandl does not teach an interface that includes graphical elements that define input/state pairs. Brandl does not teach receiving state transition data associated with one or more such graphical elements via an input device. Finally, Brandl does not teach receiving state transition data that defines a next state to which the state machine transitions when the current state of the state machine and the inputs to the state machine correspond to a particular input/state pair.

Because Brandl does not teach any of these features of the claimed invention, claims 1-79 are not anticipated under 35 U.S.C. §102(e) and should be allowed.

### C. Claims 1 to 79 are not unpatentable under 35 U.S.C. §103(a) over U.S. Patent No. 6,834,370 to Brandl et al. in view of U.S. Patent No. 5,903,886 to Heimlich et al.

To establish prima facie obviousness of a claimed invention, all of the limitations must be taught or suggested in the prior art. *In Re Royka*, 490 F.2d 981 180 USPQ 580 (CCPA 1974). Claims 1-74 of the present application are not unpatentatble under 35 U.S.C. § 103(a) because the prior art does not teach or suggest every element of any of the pending claims. As described above with regard to the rejections under 35 USC §102 (e), Brandl does not teach all of the features of any of the independent claims. Heimlich is cited as teaching a process that can be composed of a series of tasks (Heimlich, col. 3, lines 55-67; col. 9, lines 1-67; col. 10 lines 1-40; and col. 24, lines 1-67.) Heimlich is further cited as teaching a process that can be composed of series of tasks, where the process flows specifically contain an input pair into a state machine. (Figs. 9, 10 a-c, 11 a-c, and 13.) Applicants argue that the cited passages and figures from Heimlich do not teach the features attributed to them by the Examiner. However, even assuming arguendo that Heimlich does in fact teach such features, Heimlich nonetheless fails to disclose the features of the independent claims absent from the teaching of Brandl as described above. Thus, even when Brandl and Heimlich are combined, they fail to teach each and every element of any of the claims pending in the present application. Therefore the claims are non-obvious over Brandl and Heimlich and should be allowed.

## CONCLUSION

In light of the arguments given above Applicants respectfully submit that all of the claims pending in the application are in condition for allowance. Applicants therefore request that the Board reverse the final rejection of claims 1-79 and allow the application to issue.

Dated: September 8, 2009

Respectfully submitted,

By: Jeffrey H. Canfield #38,404/
Jeffrey H. Canfield
Registration No.: 38,404
MARSHALL, GERSTEIN & BORUN LLP
233 S. Wacker Drive, Suite 6300
Sears Tower
Chicago, Illinois 60606-6357
(312) 474-6300
Attorney for Applicant

## APPENDIX A

## CLAIMS

1. (Currently Amended) A method for configuring a state machine implemented in a function block associated with a process plant via a computing device having a display device and an input device, wherein the state machine defines a plurality of states and wherein the state machine transitions between states based on state machine configuration data and one or more state machine inputs, wherein the state machine inputs are associated with operation of the process plant, the method comprising:

providing a graphical user interface displayed by the display device, the graphical user interface including a plurality of graphical elements defining input/state pairs;

receiving state transition data associated with one or more of the plurality of graphical elements via the input device, the state transition data identifying one or more next states to which the state machine transitions following conditions in the process plant corresponding to the input/state pairs defined by the one or more of the plurality of graphical elements; and

storing the state transition data on a first computer readable medium associated with the function block.

2. (Currently Amended) A method according to claim 1, wherein the plurality of graphical elements comprises a first plurality of cells associated with the function block, wherein each cell of the first plurality of cells corresponds to an input/state pair; and

**APPENDIX A**
*(Continued)*

wherein receiving the state transition data comprises receiving data associated with one or more of the first plurality of cells via the input device, wherein the data is indicative of one or more next states to which the state machine is to transition when the state machine is in the state corresponding to one of the one or more of the first plurality of cells and when the input corresponding to the one or more of the first plurality of cells is a particular value.

3. (Currently Amended) A method according to claim 2, further comprising displaying the first plurality of cells on the display device and displaying indications of the state transition data in the one or more of the first plurality of cells for which state transition data have been received.

4. (Currently Amended) A method according to claim 2, wherein displaying the first plurality of cells on the display device comprises displaying a matrix comprising at least one row of cells and a plurality of columns of cells, wherein each row of cells is associated with a state machine input, and wherein each column of cells is associated with one of the plurality of states.

5. (Currently Amended) A method according to claim 2, wherein displaying the first plurality of cells on the display device comprises displaying a matrix comprising a plurality of rows of cells and at least one column of cells, wherein each row of cells is associated with one of the plurality of states, and wherein each column of cells is associated with a state machine input.

6. (Currently Amended) A method according to claim 2, wherein the particular value of the input is one of a logical one, a logical zero, a logical TRUE or a logical FALSE value.

**APPENDIX A**
*(Continued)*

7. (Currently Amended) A method according to claim 2, further comprising:

receiving data, via the input device, indicative of a number of the state machine inputs; and

identifying cells in the first plurality of cells based on the number of state machine inputs.

8. (Currently Amended) A method according to claim 7, further comprising:

receiving data, via the input device, indicative of a number of states in the plurality of states;

wherein identifying the cells in the first plurality of cells comprises identifying cells based on the number of inputs and the number of states.

9. (Currently Amended) A method according to claim 2, further comprising:

receiving data, via the input device, indicative of a number of states in the plurality of states; and

identifying a number of cells in the first plurality of cells based on the number of states.

10. (Currently Amended) A method according to claim 2, wherein the plurality of graphical elements further comprises a second plurality of cells associated with the function block, wherein each cell of the second plurality of cells corresponds to a respective one of a plurality of outputs of the function block and a respective one of the plurality of states of the state machine;

receiving output configuration data associated with one or more cells of the second plurality of cells via the input device, wherein respective output configuration data associated with each of the one or more cells of the second plurality of cells includes data indicative of an output value of the output corresponding to the cell when the state machine is in the state corresponding to the cell; and

storing the output configuration data on a second computer readable medium associated with the function block.

11. (Original) A method according to claim 10, wherein the first computer readable medium comprises the second computer readable medium.

12. (Original) A method according to claim 1, further comprising:

receiving data indicative of how to handle inputs that have a BAD status; and

storing the data indicative of how to handle inputs that have the BAD status.

13. (Currently Amended) A method according to claim 1, wherein the one or more state machine inputs comprises a plurality of state machine inputs, the method further comprising:

receiving data, via the graphical user interface, indicative of priorities associated with the plurality of state machine inputs; and

storing the data indicative of how to handle inputs that have a BAD status.

14. (Currently Amended) A method according to claim 1, further comprising receiving data indicative of whether one or more, if any, one or more state machine inputs should be ignored by the state machine; and

storing the data indicative of whether one or more, if any, one or more state machine inputs should be ignored by the state machine.

**APPENDIX A**
*(Continued)*

15. (Currently Amended) A method according to claim 1, wherein the one or more state machine inputs is to be associated with at least one of a process control system, a simulation of a process control system, a safety system, and a simulation of a safety system.

16. (Currently Amended) A method according to claim 1, wherein the one or more state machine inputs is to be received from at least one other function block associated with the process plant.

17. (Currently Amended) A method according to claim 1, wherein the one or more state machine inputs is to be received from an operator interface.

18. (Currently Amended) A tangible medium storing machine readable instructions comprising:

first code to provide a graphical user interface via a display device for configuring state machine transitions among a plurality of states, the graphical user interface including a plurality of graphical elements representing state machine input/state pairs which can be used to indicate desired transitions between states;

second code to receive state transition data identifying a state machine next state associated with one of the graphical elements via the graphical user interface; and

third code to store the state transition data on a computer readable medium associated with a function block implementing a state machine in a process plant such that the state machine transitions to the next state when conditions in the process plant correspond to the input/state pair associated with the graphical element.

19. (Currently Amended) A tangible medium according to claim 18, wherein the plurality of graphical elements comprises a first plurality of cells associated with

## APPENDIX A

*(Continued)*

the function block, wherein each cell of the first plurality of cells corresponds to a state machine input/state pair; and

wherein the second code comprises fourth code to receive respective data associated with one or more of the first plurality of cells via an input device of the computing device, wherein the respective data is indicative of a next state to which the state machine is to transition when the state machine is in the state corresponding to the cell and when the input corresponding to the cell is a particular value.

20. (Original) A tangible medium according to claim 19, further comprising fifth code to display on the display device indications of the state transition data in appropriate cells of the first plurality of cells.

21. (Currently Amended) A tangible medium according to claim 19, wherein the first code comprises fifth code to display on the display device a matrix comprising the first plurality of cells, the matrix comprising at least one row of cells and a plurality of columns of cells, wherein each row of the at least one row corresponds to a state machine input, and wherein each column of the plurality of columns is associated with a state machine state.

22. (Currently Amended) A tangible medium according to claim 19, wherein the first code comprises fifth code to display on the display device a matrix comprising the first plurality of cells, the matrix comprising a plurality of rows of cells and at least one column of cells, wherein each row of the plurality of rows corresponds to a state machine, and wherein each column of the at least one column is associated with a state machine input.

**APPENDIX A**
*(Continued)*

23. (Currently Amended) A tangible medium according to claim 19, wherein the particular value is one of a logical one, a logical zero, a logical TRUE value or a logical FALSE value.

24. (Currently Amended) A tangible medium according to claim 19, further comprising:

fifth code to receive data, via the input device, indicative of a plurality of state machine inputs; and

sixth code to determine a number of cells in the first plurality of cells based on the number of inputs.

25. (Currently Amended) A tangible medium according to claim 24, further comprising:

seventh code to receive data, via the input device, indicative of a number of state machine states;

wherein the sixth code comprises code to determine the number of cells based on the number of state machine inputs and the number of state machine states.

26. (Currently Amended) A tangible medium according to claim 19, further comprising:

fifth code to receive data, via the input device, data indicative of a number of state machine states; and

sixth code to determine a number of cells in the first plurality of cells based on the number of state machine states.

27. (Original) A tangible medium according to claim 19, wherein the plurality of graphical elements comprises a second plurality of cells associated with the

**APPENDIX A**
*(Continued)*

function block, wherein each cell of the second plurality of cells corresponds to a respective one of a plurality of outputs of the function block and a respective one of the plurality of states of the state machine;

sixth code to receive output configuration data associated with at least some of the second plurality of cells via the input device, wherein respective output configuration data associated with each cell of the at least some of the second plurality of cells includes data indicative of an output value of the output corresponding to the cell when the state machine is in the state corresponding to the cell; and

seventh code to store the output configuration data.

28. (Currently Amended) A tangible medium according to claim 18, wherein the state machine inputs comprise a plurality of inputs, the tangible medium further comprising:

fourth code to receive data, via the input device, indicative of priorities associated with the plurality of state machine inputs; and

fifth code to store the data indicative of the priorities.

29. (Currently Amended) A tangible medium according to claim 18, further comprising:

fourth code to receive data indicative of how to handle state machine inputs that have a BAD status; and

fifth code to store the data indicative of how to handle state machine inputs that have a BAD status.

30. (Currently Amended) A tangible medium according to claim 18, further comprising:

**APPENDIX A**
*(Continued)*

fourth code to receive data indicative of whether to ignore one or more, if any, of the state machine inputs; and

fifth code to store the data indicative of whether to ignore one or more, if any, of the state machine inputs.

31. (Currently Amended) A tangible medium according to claim 18, wherein at least one state machine input is to be associated with at least one of a process control system, a simulation of a process control system, a safety system, or a simulation of a safety system.

32. (Currently Amended) A tangible medium according to claim 18, wherein at least one state machine input is to be received from at least one other function block associated with the process plant.

33. (Currently Amended) A tangible medium according to claim 18, wherein at least one state machine input is to be received from an operator interface.

34. (Currently Amended) A method of implementing a state machine in a function block for use in controlling, or simulating control of, one or more field devices in a process plant, the method comprising:

providing a graphical user interface displayed by a display device, the graphical user interface including a plurality of graphical elements for configuring state machine transitions between a plurality of state machine states, the graphical elements defining one or more state machine input/state pairs, wherein one or more state machine inputs are indicative of one or more conditions within the process plant;

**APPENDIX A**
*(Continued)*

receiving state transition data identifying a state machine next state associated with at least one of the graphical elements via an interface input associated with the graphical user interface;

storing the state transition data on a first computer readable medium associated with the function block;

receiving at least one state machine input;

determining a state machine next state based on the at least one input, a current state, and the state transition data stored on the first computer readable medium;

setting the current state of the state machine to the state machine next state; and

providing at least one function block output for use in controlling the one or more field devices to at least a second other function block, wherein the at least one function block output is based on the current state of the state machine.

35. (Currently Amended) A method according to claim 34, wherein the one or more state machine inputs comprises a plurality of state machine inputs;

wherein determining the state machine next state is further based on priorities associated with the plurality of state machine inputs.

36. (Currently Amended) A method according to claim 35, wherein the state machine next state is based on an order determined by the priorities associated with the plurality of state machine inputs.

37. (Currently Amended) A method according to claim 34, further comprising:

**APPENDIX A**
*(Continued)*

determining whether a state transition is to occur based on the received at least one state machine input and the state transition data stored on the first computer readable medium;

wherein determining the next state comprises determining the next state if a state transition is to occur; and

wherein setting the current state of the state machine to the next state comprises setting the current state of the state machine to the next state if a state transition is to occur.

38. (Currently Amended) A method according to claim 34, wherein determining the next state comprises determining whether one or more, of the received at least one state machine inputs is a particular value.

39. (Currently Amended) A method according to claim 38, wherein determining the next state further comprises determining whether one or more, of the one or more of the received at least one state machine inputs that are a particular value and that also correspond to state changes based on the state transition data stored on the first computer readable medium.

40. (Currently Amended) A method according to claim 39, further comprising selecting one of the one or more, of the received at least one inputs that are a particular value and that correspond to state changes.

41. (Currently Amended) A method according to claim 40, wherein the at least one input comprises a plurality of inputs;

wherein selecting one of the one or more, of the received at least one inputs that are a particular value and that correspond to state changes comprises selecting one of the one or more of the received at least one inputs that are a particular value based on priorities associated with the plurality of inputs.

42. (Currently Amended) A method according to claim 41, wherein selecting one of the one or more, of the received at least one inputs that are a particular value and that correspond to state changes comprises selecting one of the one or more of the received at least one inputs that are a particular value based on an order associated with the plurality of inputs.

43. (Currently Amended) A method according to claim 34, wherein determining the next state comprises determining one or more, of the received at least one inputs associated with potential state changes from the current state based on the state transition data stored on the first computer readable medium.

44. (Currently Amended) A method according to claim 43, wherein determining the next state further comprises determining one or more, of the one or more of the received at least one inputs associated with potential state changes from the current state that are a particular value.

45. (Currently Amended) A method according to claim 34, wherein providing the at least one function block output comprises providing a plurality of function block outputs.

46. (Original) A method according to claim 45, wherein each of at least some of the plurality of function block outputs are indicative of whether the current state of the state machine corresponds to a respective one of a plurality of possible states of the state machine.

**APPENDIX A**
*(Continued)*

47. (Original) A method according to claim 45, wherein providing the plurality of function block outputs comprises:

retrieving, based on at least the current state, data indicative of appropriate values for at least some of the plurality of state machine function block outputs from an output configuration database; and

setting the at least some of the plurality of function block outputs to the appropriate values.

48. (Original) A method according to claim 45, wherein providing the plurality of function block outputs comprises providing one function block output indicative of the current state of the state machine.

49. (Currently Amended) A method according to claim 34, wherein the at least one function block output comprises a function block output that is indicative of the current state of the state machine.

50. (Original) A method according to claim 34, further comprising:

receiving an input indicative of whether the state machine function block is to be disabled; and

if the input indicative of whether the state machine function block is to be disabled indicates that the state machine function block is to be disabled, setting the current state of the state machine to a disabled state.

51. (Original) A method according to claim 50, further comprising:

receiving an input indicative of whether the state machine function block is to forced to an initial state; and if the input indicative of whether the state machine function block is to be forced to the initial state indicates that the state machine function block should be forced to the initial state, setting the current state of the state machine to the initial state;

wherein the input indicative of whether the state machine function block is to be enabled and the input indicative of whether the state machine function block is to be forced to the initial state comprise a single input.

52. (Original) A method according to claim 34, further comprising:

receiving an input indicative of whether the state machine function block is to forced to an initial state; and

if the input indicative of whether the state machine function block is to be forced to the initial state indicates that the state machine function block should be forced to the initial state, setting the current state of the state machine to the initial state.

53. (Currently Amended) A method according to claim 34, wherein receiving at least one state machine input comprises receiving at least one signal associated with at least one of a process control system, a simulation of a process control system, a safety system, and a simulation of a safety system.

54. (Currently Amended) A method according to claim 34, wherein the at least one state machine input is to be received from at least one other function block associated with the process plant.

55. (Currently Amended) A method according to claim 34, wherein the at least one state machine input is to be received from an operator interface.

56. (Original) A method according to claim 34, wherein providing the at least one function block output comprises providing the at least one function block output to

a portion of a process control system, wherein the portion of the process control system controls the one or more field devices based, at least in part, on the at least one function block output.

57. (Original) A method according to claim 34, wherein providing the at least one function block output comprises providing the at least one function block output to a portion of a safety system, wherein the portion of the safety system controls the one or more field devices based, at least in part, on the at least one function block output.

58. (Currently Amended) A function block entity for use in a process plant having a processor adapted to control, or to simulate control of, one or more field devices, the function block entity comprising:

a user modifiable state machine configuration database including state transition data indicative of how a state machine implemented by the function block is to transition among a plurality of states, wherein the state transition data comprises data for potential pairings of state machine states and one or more corresponding function block inputs, the state transition data indicative of a next state to which the state machine is to transition when the state machine is in a state corresponding to a particular one of the pairings and when the input corresponding to the particular one of the pairings is a particular value;

a first computer readable medium;

first code stored on the first computer readable medium to receive the inputs to the function block, wherein inputs comprise data associated with the process plant;

second code stored on the first computer readable medium to determine a next state of the state machine, wherein the determination is based on the at least one input, a

**APPENDIX A**
*(Continued)*

current state of the state machine, and the state transition data, wherein the second code is fixed;

third code stored on the first computer readable medium to set the current state of the state machine to the next state, wherein the third code is fixed; and

fourth code stored on the first computer readable medium to provide at least one function block output for use in controlling the one or more field devices.

59. (Original) A function block entity according to claim 58, wherein the state machine configuration database is stored on the first computer readable medium.

60. (Original) A function block entity according to claim 58, wherein the state machine configuration database is stored on a second computer readable medium different from the first computer readable medium.

61. (Currently Amended) A function block entity according to claim 58, wherein the one or more corresponding function block inputs comprise a plurality of inputs;

wherein the second code comprises fifth code stored on the first computer readable medium to determine the next state further based on priorities associated with the plurality of inputs.

62. (Original) A function block entity according to claim 61, wherein the fifth code comprises code to determine the next state further based on an order associated with the plurality of inputs.

63. (Currently Amended) A function block entity according to claim 58, further comprising:

fifth code stored on the first computer readable medium to determine whether a state transition is to occur based on at least one of the one or more corresponding function block inputs and the state transition data;

wherein the second code comprises code to determine the next state if a state transition is to occur; and

wherein the third code comprises code to set the current state of the state machine to the next state if a state transition is to occur.

64. (Currently Amended) A function block entity according to claim 58, wherein the second code comprises fifth code stored on the first computer readable medium to determine which, if any, of the inputs to the function block are the particular value.

65. (Currently Amended) A function block entity according to claim 64, wherein the second code further comprises sixth code stored on the first computer readable medium to determine which, if any, of the inputs to the function block are the particular value and which also correspond to a state change based, at least in part, on the state transition data.

66. (Currently Amended) A function block entity according to claim 65, further comprising seventh code stored on the first computer readable medium to select one, if any, of the inputs to the function block that are the particular value and that correspond to a state change, that corresponds to a state change from the current state.

67. (Currently Amended) A function block entity according to claim 66, wherein the one or more function block inputs comprise a plurality of inputs;

wherein the seventh code comprises eighth code stored on the first computer readable medium to select the one of the function block inputs based on priorities associated with the plurality of function block inputs.

## APPENDIX A

*(Continued)*

68. (Currently Amended) A function block entity according to claim 67, wherein the eighth code comprises code to select one of the function block inputs of the plurality of function block inputs based on an order associated with the plurality of function block inputs.

69. (Currently Amended) A function block entity according to claim 58, wherein the second code comprises fifth code stored on the first computer readable medium to determine one or more, if any, of the one or more function block inputs that would cause a state change from the current state based on the state transition data stored on the second computer readable medium.

70. (Currently Amended) A function block entity according to claim 69, wherein the second code further comprises sixth code stored on the first computer readable medium to determine whether one or more, if any, of the one or more function block inputs that would cause a state change is also the particular value.

71. (Original) A function block entity according to claim 58, wherein the fourth code comprises fifth code stored on the first computer readable medium to provide a plurality of function block outputs.

72. (Currently Amended) A function block entity according to claim 71, wherein the fifth code comprises code to provide each of at least some of the plurality of function block outputs, wherein the lat least some of the plurality or function block outputs are indicative of whether the current state of the state machine corresponds to a respective one of a plurality of possible states of the state machine.

73. (Original) A function block entity according to claim 71, wherein the fifth code comprises:

**APPENDIX A**
*(Continued)*

sixth code stored on the first computer readable medium to retrieve, based on at least the current state, data indicative of appropriate values for at least some of the plurality of state machine function block outputs from a user configurable output configuration database; and

seventh code stored on the first computer readable medium to set the at least some of the plurality of function block outputs to the appropriate values.

74. (Original) A function block entity according to claim 73, wherein the state machine configuration database and the output configuration database are stored on a same computer readable medium.

75. (Original) A function block entity according to claim 73, wherein the state machine configuration database and the output configuration database are stored on different computer readable media.

76. (Original) A function block entity according to claim 58, further comprising:

fifth code stored on the first computer readable medium to receive an input indicative of whether the state machine function block is to be disabled; and

sixth code stored on the first computer readable medium to set the current state of the state machine to a disabled state if the input indicative of whether the state machine function block is to be disabled indicates that the state machine function block is to be disabled.

77. (Currently Amended) A function block entity according to claim 58, further comprising:

**APPENDIX A**
*(Continued)*

fifth code stored on the first computer readable medium to receive an input indicative of whether the state machine function block is to be forced to an initial state; and

sixth code stored on the first computer readable medium to set the current state of the state machine to the initial state if the input indicative of whether the state machine function block is to be forced to the initial state indicates that the state machine function block should be forced to the initial state.

78. (Currently Amended) A function block entity according to claim 58, wherein the inputs to the function block comprise at least one signal associated with at least one of a process control system, a simulation of a process control system, a safety system, or a simulation of a safety system.

79. (Currently Amended) A function block entity according to claim 58, further comprising fifth code stored on the first computer readable medium to mask one or more, if any, of the inputs to the function block.

## APPENDIX B

No evidence pursuant to §§ 1.130, 1.131, or 1.132 or entered by or relied upon by the examiner is being submitted.

# APPENDIX C

No related proceedings are referenced in II. above, hence copies of decisions in related proceedings are not provided.